

[19]中华人民共和国国家知识产权局

[51]Int. Cl⁷

G06F 12/02
G06F 12/16

[12] 发明专利申请公开说明书

[21] 申请号 98802593.0

[43]公开日 2000 年 3 月 22 日

[11]公开号 CN 1248334A

[22]申请日 1998.12.8 [21]申请号 98802593.0
[30]优先权
[32]1997.12.16 [33]JP [31]363504/97
[32]1997.12.22 [33]JP [31]365704/97
[86]国际申请 PCT/JP98/05545 1998.12.8
[87]国际公布 WO99/31592 日 1999.6.24
[85]进入国家阶段日期 1999.8.16
[71]申请人 TDK 株式会社
地址 日本东京都
[72]发明人 柿沼裕二 北川浩也 堀田辉男

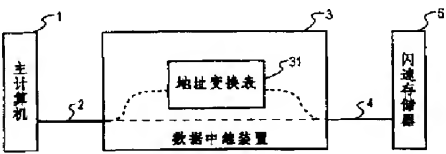
[74]专利代理机构 中国国际贸易促进委员会专利商标事
务所
代理人 王永刚

权利要求书 5 页 说明书 22 页 附图页数 10 页

[54]发明名称 闪速存储器系统

[57]摘要

本发明提供具有管理主计算机与闪速存储器的数据
传送的存储器管理器的闪速 存储器系统。存储器管理
器具有在从上述主计算机提供给闪速存储器的逻辑地
址与作为闪速存储器的实际地址的物理地址之间进行变
换的地址变换表。而且,地址变换表具有规定为对应于
闪速存储器的最小消除单位的结构。使用这 样的结构,
能够实现写入/读出时间延迟少,能够以高速进行动作,
能够适宜 地进行不良扇区和不良比特等的管理的闪速
存储器系统。



ISSN 1000-8427 4

时，与物理地址区中的不与逻辑地址区对应部分的最小消除单位的任一个进行交换。

7. 如权利要求 1 至 6 的任一项中记述的闪速存储器系统，特征在于：

上述数据中继装置还具有块状态表，

该块状态表是表示闪速存储器上的上述各物理块的状态的数据，

至少具有表示其块是否良好，有无不良扇区以及是否正在使用的数据。

8. 如权利要求 1 至 7 的任一项中记述的闪速存储器系统，特征在于：

上述地址变换表用在复位以后从闪速存储器的冗余部分读出的数据形成在 RAM 上。

9. 如权利要求 7 或 8 中记述的闪速存储器系统，特征在于：

上述块状态表用在复位以后从闪速存储器的冗余部分读出的数据形成在 RAM 上。

10. 一种闪速存储器系统，特征在于：

具有管理主计算机与闪速存储器的数据传送的存储器管理器，

上述存储器管理器把从上述主计算机能够存取闪速存储器的逻辑地址区的大小设定为小于作为闪速存储器的实际地址的物理地址区的大小，

而且在物理地址区不与逻辑地址区对应的剩余区域中具有用于待机状态的队列。

11. 如权利要求 10 中记述的闪速存储器系统，特征在于：

上述存储器管理器在对应闪速存储器的最小消除单位分割了的每个块中处理上述物理地址区，上述逻辑地址区以及上述队列。

12. 如权利要求 10 或 11 中记述的闪速存储器系统，特征在于：对应于上述队列的闪速存储器的区域成为消除后的状态。

13. 如权利要求 10 至 12 的任一项中记述的闪速存储器系统，

特征在于:

上述存储器管理器在物理地址区内任意的块中发生了故障时,与上述队列内的任意的块进行交换。

14. 如权利要求 10 至 13 的任一项中记述的闪速存储器系统,特征在于:

在向闪速存储器写入数据时,上述存储器管理器向队列内的预定的块中写入新的数据,

上述存储器管理器把该数据设为写入预定目标的逻辑块地址,把该写入预定目标的块设定为队列。

15. 如权利要求 14 中记述的闪速存储器系统,特征在于:

在写入预定目标的块具有将被更新的数据以外的数据的情况下,

在结束了来自主计算机的预定量的数据传送以后,

上述存储器管理器从写入预定目标的块向被写入了新的数据的队列内的预定的块中传送原来的数据。

16. 如权利要求 10 至 15 的任一项中记述的闪速存储器系统,特征在于:

上述存储器管理器使得物理地址区和逻辑地址区与对应于最小消除单位而分割的逻辑块地址和物理块地址相互对应,

而且对应关系的开始位置取为由每个预定次数的复位所决定的任意位置。

17. 如权利要求 10 至 16 的任一项中记述的闪速存储器系统,特征在于:

上述存储器管理器在每个预定次数的复位时,在队列的块中写入对应于逻辑地址区的物理地址区内的任意块中的数据,

而且,把写入了数据的队列内的块地址设为物理地址区内的任意块的逻辑块地址,

把上述物理地址区内的任意的块设为队列。

18. 如权利要求 10 至 17 的任一项中记述的闪速存储器系统,

特征在于:

除去上述存储器管理器以外, 还具有用于与主计算机连接的总线接口, 进行向闪速存储器的读出/写入动作的闪速存储器接口。

19. 如权利要求 10 至 18 的任一项中记述的闪速存储器系统, 特征在于:

上述队列用指针控制。

20. 如权利要求 1 至 19 的任一项中记述的闪速存储器系统, 特征在于:

上述最小消除单位相当于 8 或者 16 个最小存储单位。

21. 如权利要求 1 至 20 的任一项中记述的闪速存储器系统, 特征在于:

上述最小存储单位是在一个扇区部分的数据长度上加入了冗余部分的数据长度的大小。

22. 如权利要求 1 至 21 的任一项中记述的闪速存储器系统, 特征在于:

上述最小存储单位的冗余部分中至少具有表示最小存储单位所属的块是否良好, 有无不良扇区以及其块是否正在使用的数据。

23. 如权利要求 1 至 22 的任一项中记述的闪速存储器系统, 特征在于:

上述冗余部分在一个块中存储着相同的数据。

24. 如权利要求 1 至 23 的任一项中记述的闪速存储器系统, 特征在于:

上述数据中继装置还具有暂时存储预定长度的数据的 2 个缓冲器, 在这 2 个缓冲器中交互地存储从主计算机一侧或者闪速存储器一侧发送来的数据的同时, 从装满了数据的缓冲器交互地把数据送出到闪速存储器一侧或者主计算机一侧。

25. 如权利要求 1 至 24 的任一项中记述的闪速存储器系统, 特征在于:

该系统是 IC 芯片。

26. 如权利要求 1 至 25 的任一项中记述的闪速存储器系统，特征在于：

该系统是卡片形状的外部存储系统。

闪速存储器系统

技术领域

本发明涉及作为非易失性存储器的闪速存储器，更详细地讲涉及具备适宜地管理闪速存储器的管理系统的闪速存储器系统。

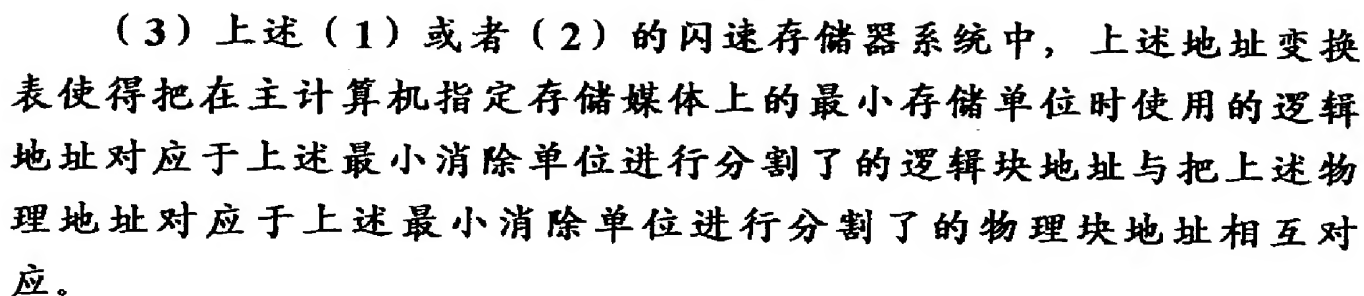
背景技术

作为计算机系统所使用的存储媒体，当前正在使用着硬盘，软盘等磁存储媒体。其中，大容量而且能够快速动作的硬盘由于能够存储大量的数据，因此处于系统的中心存储媒体的位置。

然而，硬盘的制造工艺复杂，难以谋求小型，轻量化和低成本化，为了使构造的各部分工作而需要比较大的消耗功率，特别是在便携设备等的应用方面成为问题。

作为硬盘以外的存储媒体，已知作为非易失性的存储元件的闪速存储器。闪速存储器不需要用于维持存储的电源，然而其构造上，在存取次数方面具有 10^{10} 左右的有限的物理寿命。为此，为了长期比较安全地稳定使用闪速存储器，需要检测寿命完结的存储元件和发生不良状况的位置，回避这些元件和位置保持数据。进而，在闪速存储器的情况下，不能够在已经存储的数据的位置上写新的数据。从而，在要存储新的数据时，要在消除了暂时存储的内容以后写入数据。然而，可消除的单位不是以一个个的存储元件（比特），而是例如以 4k 字节和 8k 字节这样的块单位进行。因此，通常把需改写的的数据以外的数据部分从将被消除的块中取出之后，再对块进行消除，然后，再次把数据部分重新加入或写入，写入处理动作非常复杂。

在特开平 2-292798 号公报中记述了以提高向闪速存储器的写入/读出速度，管理不良单元等为目的的闪速 EEPROM 系统。在这



(4) 上述(1) - (3)的任一项中的闪速存储器系统中, 上述逻辑地址与物理地址的变换在向闪速存储器开始数据的写入/读出之前进行。

(5) 上述(1)~(4)的任一项中的闪速存储器系统中, 上述闪速存储器的物理地址区的大小大于逻辑地址区的大小。

(6) 上述(1)~(5)的任一项中的闪速存储器系统中, 在对应于上述逻辑地址区的任意的最小消除单位中发生了故障时, 与物理地址区中的不与逻辑地址区对应的部分的任一个最小消除单位进行交换。

(7) 上述(1)~(6)的任一项中的闪速存储器系统中, 上述数据中继装置还具有块状态表, 该块状态表是表示闪速存储器上的上述各物理块的状态的数据,

至少具有表示其状态是否良好，有无不良扇区以及是否使用的数据。

(8) 上述(1)~(7)的任一项中的闪速存储器系统中, 上述地址变换表用在复位以后从闪速存储器的冗余部分读出的数据形成在RAM上。

(9) 上述(7)或者(8)的闪速存储器系统中, 上述块状态表用在复位以后从闪速存储器的冗余部分读出的数据形成在 RAM 上。

(10) 閃速存儲器系統，具有管理主計算機與閃速存儲器的數據傳送的存儲器管理器，

上述存储器管理器把能够从上述主计算机存取闪速存储器的逻辑地址区的大小设定为小于作为闪速存储器的实际地址的物理地址

图 1 是示出本发明的闪速存储器系统的基本结构的框图。

图 2 是示出地址变换表一结构例的示意图。

图 3 是示出闪速存储器内的块结构的示意图。

图 4 是示出存储器管理器内理想地具有的数据中继装置结构例的框图。

图 5 是示出块状态表的结构例的示意图。

图 6 是示出本发明的闪速存储器系统的基本结构的框图，示出在闪速存储器内设置了剩余区域的状况。

图 7 是对于闪速存储器内的物理地址区配置了逻辑地址的示意图，示出对应于逻辑地址区的物理地址区。

图 8 示出了剩余区域分散配置在物理地址区中的状况。

图 9 示意地示出了队列的状况。

图 10 是示出了闪速存储器内的实际数据存储部分与冗余数据存储部分的状况的示意图。

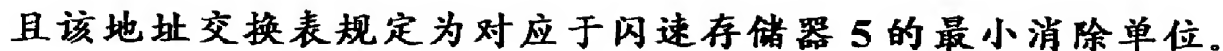
图 11 是复位时的动作，是示出了形成块状态表的状况的示意图。

图 12 是复位时的动作，是示出了在闪速存储器中进行临时编号状况的示意图。

图 13 是示出把闪速存储器系统应用在存储器卡中的例子的结构框图。

用于实施发明的最佳形态

本发明的闪速存储器系统如图 1 所示，具有存储器管理器 3，该存储器管理器 3 具有能够与主计算机 1 连接的总线接口 2，用于进行在闪速存储器 5 上写入/读出的闪速存储器接口 4，用于把从上述主计算机一侧向闪速存储器一侧发送来的数据和把从闪速存储器一侧向主计算机一侧发送的数据进行中继的数据中继装置，上述存储器管理器 3 具有把从主计算机 1 提供给闪速存储器 5 的逻辑地址和作为闪速存储器 5 的实际地址的物理地址进行交换的地址交换表，而



存储器管理器 3 管理主计算机 1 与闪速存储器 5 的数据传送。即，或者调整从主计算机 1 发送来的数据和从闪速存储器 5 发送来的数据的传送速度，或者把用于从主计算机 1 指定的读出/写入的逻辑地址数据变换为物理地址数据。

地址变换表 31 把在主计算机 1 一侧指定的逻辑地址与闪速存储器 5 一侧存在的物理地址之间建立关系。通过借助地址变换表 31, 把逻辑地址与物理地址之间建立关系, 能够避开在闪速存储器内产生了不良的位置等而把两者建立关系。而且, 不必特意从闪速存储器读出不良扇区和不良比特等的的数据, 事先进行适宜的关系对应, 在存取存储器时参照该关系能够立即计算必要的物理地址, 高速地存取闪速存储器。

地址变换表 31 设置在存储器管理器 3 内。另外，通常形成在从该存储器管理器 3 的控制单元可参照的存储媒体上。而且，存储器管理器 3 使用该地址变换表 31，把从主计算机 1 提供给闪速存储器 5 的逻辑地址变换为作为闪速存储器 5 的实际地址的物理地址，或者进行其逆变换。

主计算机 1 具有微处理器以及该微处理器的周边元件，键盘，通信口，扩展总线，显示器等输入输出装置，只要是能够与硬盘等的存储媒体进行连接，可以作为计算机（个人计算机）完成必要的动作（无论一般用途还是特定用途）即可，而不论其形态如何。

总线接口 2 是用于与主计算机 1 连接的总线，例如可以举出 SCSI, IDE 等的外部连接总线，以这些总线为基准的 PCMCIA 等 PC 卡接口等。其中，作为理想的形态，在把本发明的闪速存储器系统应用在 PC 卡的情况下，使用 PC 卡接口。从而，总线接口是能够根据这些规格，标准适宜地与主计算机 1 连接的接口。另外，该总线接口如果在从主计算机 1 一侧观看的情况下，能够把闪速存储器 5 与硬盘进行相同的处理，则就能够直接使用现存的 OS，应用软件等。

闪速存储器 5 是非易失性存储器，可以视为 EEPROM 的一种。即，能够随时进行数据的读出，然而数据的写入通常在数据消除后仅能够进行一次，不能够进行数据的改写。从而，进行数据的写入动作之前必须伴随着用于消除存储着的数据的动作（除去初始状态）。在消除数据的情况下，在闪速存储器中，其构造上成为一起消除预定的存储区域。本发明中理想地使用的 NAND 型闪速存储器中，通常，可消除的最小区域是上述最小存储单位（一个扇区 + 冗余部分）的 8 个或者 16 个部分，即，4k 字节或者 8k 字节（除去冗余部分）。

闪速存储器 5 为了确保预定的存储容量，通常构成为集合了多个闪速存储器 IC 的存储器阵列。另外，还连接了与闪速存储器 5 相连接的，用于对闪速存储器 5 进行写入/读出动作的闪速存储器接口 4，例如，闪速存储器序列发生器等以及自动控制和优化闪速存储器的写入/读出的控制元件等。

其次，说明地址变换表 31 的具体结构例。图 2 是示出地址变换表 31 的结构例的示意图。地址变换表 31 形成为与图 3 所示的闪速存储器 5 内的最小消除单位（最小消除块）5a 相对应。另外，图 3 示意性示出了闪速存储器 5 的方框结构，图中标注在上面的带有 # 号的数值表示为每个最小消除单位所规定的物理块地址。

最小消除单位是在闪速存储器 5 内一起被消除的存储区域。如上所述，闪速存储器 5 的最小存储单位与最小消除单位不同，消除每个预定大小的存储区域。把该一起被消除的区域称为最小消除单位，闪速存储器的消除动作必须对作为集合了多个最小存储区域的块的最小消除单位进行。另外，最小消除单位 5a 中的实线所包围的部分是最小存储单位，其中，用虚线划开的部分中较小一方的区域表示冗余部分。

这样，通过用每个最小消除单位 5a 处理并且管理消除每个最小消除单位 5a 所需要消除的闪速存储器 5，能够高效地进行写入动作（读出动作）。从而，地址变换表 31 形成为与最小存储单位相对应。

地址变换表 31 在图 2 所示的例中，在一个块内的上段配置逻辑块地址，在下段配置物理块地址（图中编号的起端标注 # 号表示），对于某个特定的逻辑块地址，通常能够把任意的物理块地址以一对一建立关系。即，相对应的逻辑块地址与物理块地址配置在预定的块内。而且，规定在主计算机 1 一侧开放的可存取的存储区的最大逻辑块地址 N_h 和具有对应大小的物理块地址。

这里所谓的逻辑块地址，是在从主计算机 1 存取特定的存储器内的存储单位时指定的地址编号，指把地址分割为对应于上述最小消除单位 $5a$ 大小的每个块而得到的起始地址。所谓物理块地址，是闪速存储器上的实际的地址编号，指最小消除单位 $5a$ 的起始地址。

用地址变换表使逻辑块地址与物理块地址两者相互对应的意义如下。由于通常逻辑块地址与物理块地址以一对一相对应，因此有时还能够直接把逻辑块地址用作为物理块地址。然而，逻辑块地址与物理块地址之间有时产生偏离（或者偏移）。这些偏差起因于两个管理系统不同，起因于由于产生故障而不能使用的不良块以及起因于为了防止在特定的存储区域集中进行存取而实施的处理等。

其中，发生了不良块的情况下，需要排除其逻辑块地址，把逻辑块地址进行分配。图 2 示出这样把逻辑块进行分配的状态。该例中，# 0003 和 # 0020 的物理块地址作为不良块被排除的结果，# 0004 成为 # 0003，# 0021 成为 # 0019，逻辑块地址的最大值 N_h 中，额外地需要 2 个块，即物理块地址 # $N_h + 2$ 。另外，在该例中将物理块地址依次分配给逻辑块地址，然而也可以分配给任意的逻辑块地址。

这样，有时在能够用逻辑块地址即逻辑地址区指定的存储区与作为实际存储区的物理地址的最大区域的物理地址区之间具有偏差，理想的是物理地址区设定为大于逻辑地址区。即，由于物理地址区中产生的故障可以被相关物理地址区中的存储区所补偿，故逻辑地址区设定为较小。换言之，在物理地址区中存在着不包含在逻辑地址区内的剩余区域。

缓冲器 B(A) 置为数据读出一侧(虚线一侧)。而且, 缓冲器控制单元 35 在作为数据写入一侧的一方的缓冲器 A(B) 装满了预定长度的数据, 而且检测出已经读出了作为数据读出一侧的另一方的缓冲器 B(A) 的数据时, 把两者进行交换, 把装满了数据的一方的缓冲器 A(B) 置为数据读出一侧, 把已经读出了数据的另一方的缓冲器 B(A) 置为数据写入一侧。

这样, 在 2 个缓冲器中交互地写入数据的同时, 通过从装满了数据的缓冲器交互地读出数据, 即使主计算机一侧与闪速存储器一侧数据的传送速度不同, 也能够对应于各个速度和定时收发数据。这种情况下, 由于数据收发速度快的一侧必须成为待机状态, 因此作为总体的数据传送速度由数据传送速度慢的一侧支配。另外, 缓冲器内的数据在被读出以后, 虽然像 FIFO 存储器那样成为空状态, 但如果能够检测出用新数据装满了的状态, 则也能够进行改写。

在 2 个缓冲器中作为能够写入/读出的数据长度, 虽然没有特别的限定, 然而理想的是作为一个扇区部分数据为 512 字节或者在其上面加入了冗余数据部分的长度。

这样, 通过交互地交换 2 个缓冲器 36、37 进行的数据收发, 能够在一方的缓冲器中写入数据的同时从另一方的缓冲器读出数据, 因此能够大幅度地缩减用于中继数据所需要的时间。这种情况下, 缓冲器 36、37 的交换由缓冲器控制单元 35 瞬时地进行, 因此如上述那样数据的传送时间能够视为与主计算机一侧或者闪速存储器一侧的某一个较慢一方的速度相同。另外, 缓冲器存储器 35 作为门电路的组合, 最好用硬件构成。通过仅采取硬件的结构能够大幅度地提高动作速度。

本发明的闪速存储器系统还可以具有块状态表 32, 该块状态表 32 具有表示闪速存储器上的各物理块的状态的数据, 至少表示该块是否良好, 有无不良扇区以及是否正在使用的数据。通过具有块状态表 32, 能够在表上迅速地把握各物理块的状态, 能够迅速地形成后述的队列, 能够在存储器的读出和写入之前或者与其无关地进行

适宜的存储器管理。

块状态表 32 例如像图 5 所示那样构成。在图示例中，在一个块内的上端配置物理块地址（图中编号的起始标注 # 号表示），在下段配置状态值，使得具有对应于各个物理块地址的状态值。然后，规定直到闪速存储器 5 内的可存取存储区中的最大物理块地址 # Nmax 的对应状态值。

状态值是表示物理块地址的状态的值，例如，作为初始状态写入 FF（或者消除状态），然后根据块地址的状态把值进行变化。作为能够用状态值表现的数据，至少是表示块是否良好，有无不良扇区以及其块是否正在使用的数据。除此之外，还能够表示块良好的程度以及有关安全数据等。作为数据的表示形态虽然没有特别的限定，然而既可以是特定的数据长度，例如以 1 个字节的代码进行表现，也可以通过使特定数据长度的各比特具有权值进行表现。

通过具有块状态表 32，能够迅速地把握各逻辑块状态。从而，在参照块状态表构筑地址变换表 31 时，排除后述的队列要素等，把逻辑块地址分配到物理块地址中，或者在有不良扇区时，迅速地检测成为不良块等的块，能够构筑将不良块排除的地址变换表。另外，还能够从块状态表 32 容易地把握未使用的块，能够极其容易而且迅速地进行与不良块的交换等的作业。即，能够不存取闪速存储器而把握各扇区和块的状态，格外地提高处理速度。

上述存储器管理器 3，例如如图 6 所示，还把能够从上述主计算机 1 存取闪速存储器 5 的逻辑地址区 Ln 的大小设定为小于作为闪速存储器的实际地址的物理地址区 Pn 的大小，而且在物理地址区 Pn 的不与逻辑地址区 Ln 相对应的剩余区域 5a 中具有使用待机状态的队列。

通过把作为闪速存储器的实际地址的物理地址区的大小设定为大于从主计算机存取闪速存储器时的逻辑地址区的大小，而且在物理地址区的不与逻辑地址区相对应的剩余区域中具有队列，例如，在与逻辑地址区相对应的物理地址区内发生了不良位置的情况下能

够与队列部分进行交换。另外，在向闪速存储器进行写入时，把新的写入数据暂时写入到队列部分中，根据需从写入的预定部位补充原来的数据，将其与写入预定部分进行交换，由此能够极其高效率的进行写入时的动作。

另外，不是简单地任意地使用上述剩余区域，而通过作为队列进行管理，在特定的存储器区中集中存取等，能够不进行复杂的控制进行适宜的存储器管理。

图 6 中，存储器管理器 3 把从上述主计算机能够存取闪速存储器的逻辑地址区的大小 L_n 设定为小于作为闪速存储器的实际地址的物理地址区的大小 P_n ，而且在物理地址区的不与逻辑地址区相对应的剩余区域，具有使用等待状态的队列。

即，作为闪速存储器 5 内的全部存储区域的物理地址区，设定为比能够从主计算机 1 一侧进行存取的逻辑地址区大。换言之，在主计算机一侧开放的逻辑地址区仅比作为闪速存储器 5 的整个存储区域的物理地址区小包含队列部分的预定区域。

队列 (Queue) 通常在初始状态从上述剩余区域中的若干地址顺序形成 (但是除去不良位置等)。另外，该队列 (Queue)，物理地址以及逻辑地址最好在每个闪速存储器 5 的最小消除单位 (块) 进行处理。该最小消除单位在闪速存储器上是同时被消除的最小的单位。如上所述，闪速存储器不是在每个最小存储单位，而是在每个把它们集合起来成为某个大小的存储单位进行消除。因此，通过在每个作为该消除区域的最小单位的最小存储单位，处理上述队列 (Queue)，物理地址以及逻辑地址，能够得到高效的存储器控制系统。

在对应于逻辑地址区的物理地址区内发生了不良位置的情况下，该不良位置与队列进行置换，而这些操作通过在每个块进行也能够高效地适宜地进行管理。闪速存储器 5 的写入动作需要每个块的消除动作，而通过在每个块管理闪速存储器 5 能够高效地进行动作。

物理地址通常当然是与逻辑地址 1 对 1 相对应，而这些对应关系理想的是在每个块中进行规定。即，两者分别在每个块中处理，对于其起始地址，规定为物理块地址和逻辑块地址。而且，作为从起始地址开始的偏移值，规定作为每个存储单位的物理地址以及逻辑地址。另外，各个块使用地址变换表等，从逻辑块地址变换为物理块地址或者从物理块地址变换为逻辑块地址。

如上所述在全部物理地址区中，队列 (Queue) 存在于与逻辑地址区相对应的物理地址区以外的区域内，但全部这些区域不必都是队列 (Queue)。剩余区域中有时存在作为不良块永久被废弃，也不用作为队列 (Queue) 的块等。另外，该剩余区域为全部物理地址区的 1~5%，更理想的是 2~3% 左右。

其次，说明队列 (Queue) 的更具体的结构。图 7 是示意地示出了闪速存储器 5 内的存储器空间的框图。图中各块内记载的数字，记号表示物理块地址。

在图示例的闪速存储器 5 中，在主计算机 1 一侧开放的物理块区与逻辑块地址对应，成为至其最大值 N_h 的范围。另外，作为剩余区域 5a (图中用斜线表示)，存在于从与逻辑块地址的最大值 N_h 相对应的物理块地址上加 1 的地址 $N_h + 1$ 开始至物理块地址的最大值 N_{max} 之间。而且在该区域内具有队列 (Queue)。

该队列 (Queue) 通常在初始设定时如上述那样从作为剩余区域定义的区域中选择。从而，队列 (Queue) 的各要素成为消除状态 (初始状态)，使用该队列 (Queue) 时能够直接进行写入。另外，在加入新的队列 (Queue) 时，在消除之后或者在确认了是消除状态以后进行加入。

另外，剩余区域 5a (队列) 如图 7 所示，不需要连续地存在于物理地址区的最后部分，例如，也可以像图 8 所示那样，孤立地分散存在于物理地址区内 (图中，用斜线表示的部分)。把剩余区域分散了的结果，在图示例中逻辑地址区的最大值 N_h 和物理地址区的最大值 N_{max} 成为相同位置。剩余区域 (队列) 的分散状态通常通

过反复地用后述的不良块和写入动作时的队列 (Queue) 进行置换操作形成。另外，图 8 中，在块中记载的数字与图 7 的情况不同表示逻辑块地址。这样，具有队列 (Queue) 的剩余区域 5a 由于不与逻辑地址相对应，因此不能够从主计算机 1 一侧进行识别。

队列 (Queue) 50 最好用指针控制。指针例如如图 9 所示进行控制使得构成队列 (Queue) 50 的各块 (以下称为要素) 的数成为预定的范围。即，在图示例中，在块上部标注作为要素的块的顺序，而在该顺序的预定位置处存在取出指针 OP (在图示例中 2) 和取入指针 IP ($n-2$)。而且，队列 (Queue) 50 的取出，取入时移动指针。即

(取入指针 IP 的位置) - (取出指针 OP 的位置) = 要素数

另外，队列 (Queue) 自身也可以登录到管理表等中进行管理。

本发明的闪速存储器系统理想的是具有用于把逻辑块地址与物理块地址相互交换的地址变换表。

图 2 所示的地址变换表 31 中，设定能够从主计算机存取的逻辑地址区使其小于作为实际存储器区且物理地址的最大区域的物理地址区。即，像可以设定上述队列 (Queue) 那样较小地设定逻辑地址区。在地址变换表中，不与逻辑块地址相对应的物理块地址可以成为队列 (Queue) 的要素。

使用地址变换表 31 当然能够在逻辑块地址与物理块地址之间建立关系，然而两者对应关系不一定必须成为从小的地址数开始的顺序，也可以以任意的地址彼此之间进行对应。即，形成预定区域内的两者的块地址之间，可以以 1 对 1 建立关系。从而，还能够不使用地址变换表而使用函数等。

通过闪速存储器系统的复位动作，把逻辑块地址再次与物理块地址建立关系的情况下，通常从小的地址编号顺序建立关系。这种情况下，对于逻辑块地址的最小地址编号，可以任意选择开始对应关系的物理块地址编号变更开始位置。通过这样做，能够不固定与特定的逻辑块地址编号对应的物理块地址编号而进行变动，例如，

在小的地址编号集中存取，或者即使存在在某个应用软件中被频繁地存取的区域，由于对应的闪速存储器的实际地址发生变化因此能够分散实际被存取的位置。

作为指定任意的开始位置的方法没有特别的限定，可以使用通过众所周知的随机数发生方法得到的随机数和函数及其它数学方法选择任意的块地址。

另外，通过闪速存储器系统的复位动作等，在与逻辑块地址相对应的物理块地址中，可以把任意的块地址与上述队列（Queue）的块进行交换。通过进行这样的交换，与上述相同能够防止对于特定的存储器的存取过于集中，能够把闪速存储器总体的存取次数平均化。

存储器块交换的频度可以在每次复位等时进行，在复位等的频度高的情况下可以记述预定次数，或者使用随机数等在每个预定的次数（频度）进行。

本发明的闪速存储器系统与闪速存储器共同使用，而作为把上述结构形成为一体的装置，还能够与闪速存储器独立地存在。例如，可以把总线接口 2，闪速存储器接口 4，地址变换表，具有数据中继装置的存储器管理系统 3 构成为一体的 IC（集成电路）芯片。通过构成 IC 芯片，在与闪速存储器组合时可以容易地得到小型的存储器系统。另外，把本发明闪速存储器系统与闪速存储器相组合，把它们配置在预定规格的卡片形状的外壳内，把总线接口作为 PC 卡总线，由此能够容易地得到不需要电源的非易失性存储器卡。

实施例

其次示出实施例，更具体地说明本发明。

[制造时的初始设定]

在制造以后的初始状态，存储器管理器 3 的控制单元对于闪存存储器 5 内的各扇区进行评价（能够正常写入/读出，被写入的数据内容中无异常等），把它们当作为块单位的评价内容制作状态表 32。



例如，如图 10 所示，假设在具有一定大小的实际数据存储单元 51 和冗余数据存储单元 52 的闪速存储器的物理块地址 # 0003 中存在异常。控制单元例如如图 11 所示，按照物理块地址顺序进行评价，在 S-RAM 上的状态表形成区域的预定位置，没有异常的情况下设定（写入）初始值（图示例中是“FF”），在检测出了异常的情况下写入表示其内容的状态值（图示例中是“0F”）。这样，制作块状态表 32。

接着，控制单元例如如图 12 所示对于闪速存储器 5 进行相当于逻辑块地址的临时编号。这时，根据状态表 32 的信息，从编号的对象消除不良块。被编号了的数值（图示例中为下段）与状态值（图示例中为上段）写入到闪速存储器的冗余部分 52 中。

在该状态的闪速存储器系统中不进行逻辑块地址与物理块地址的关系对应，也不存在队列（Queue）。然而，上述不良块成为剩余区域的一部分。

[复位时的初始设定]

开启电源时, 在复位时, 控制单元把握闪速存储器的大小 (芯片数), 求作为最小消除单位的块的总数 N_{\max} 。

接着，控制单元读出闪速存储器 5 内的各块（物理块）的冗余部分。这种情况下，在块内的各个最小存储单位（1 字节 + 冗余部分）中存在冗余部分，同一块内的冗余部分全部被写入相同的数据。从而，可以只读出每一个块的最小存储单位中的冗余部分。另外，读出并对照多个位置的冗余部分（通常为相同内容），还能够评价其最小存储单位和块的可靠性。

作为被读出的数据，上述的例中如下。

物理块地址	状态值	逻辑块地址值
0	0	0
1	0	1
2	0	2
3	0	3
4	0	4
5	0	5
6	0	6
7	0	7
8	0	8
9	0	9
10	0	10
11	0	11
12	0	12
13	0	13
14	0	14
15	0	15
16	0	16
17	0	17
18	0	18
19	0	19
20	0	20
21	0	21
22	0	22
23	0	23
24	0	24
25	0	25
26	0	26
27	0	27
28	0	28
29	0	29
30	0	30
31	0	31
32	0	32
33	0	33
34	0	34
35	0	35
36	0	36
37	0	37
38	0	38
39	0	39
40	0	40
41	0	41
42	0	42
43	0	43
44	0	44
45	0	45
46	0	46
47	0	47
48	0	48
49	0	49
50	0	50
51	0	51
52	0	52
53	0	53
54	0	54
55	0	55
56	0	56
57	0	57
58	0	58
59	0	59
60	0	60
61	0	61
62	0	62
63	0	63
64	0	64
65	0	65
66	0	66
67	0	67
68	0	68
69	0	69
70	0	70
71	0	71
72	0	72
73	0	73
74	0	74
75	0	75
76	0	76
77	0	77
78	0	78
79	0	79
80	0	80
81	0	81
82	0	82
83	0	83
84	0	84
85	0	85
86	0	86
87	0	87
88	0	88
89	0	89
90	0	90
91	0	91
92	0	92
93	0	93
94	0	94
95	0	95
96	0	96
97	0	97
98	0	98
99	0	99

# 0000	FF	0000
# 0001	FF	0001
# 0002	FF	0002
# 0003	0F	-----

# 0004	FF	0003
# 0005	FF	0004

... .. 以下省略

从所得到的数据把地址变换表 31 以及块状态变换表 32 形成在 S-RAM 上的地址变换表形成区域和状态表形成区域(因为在开启电源时或在复位时 S-RAM 上的数据消失或失去可靠性)。由此,以后直到进行再次的复位,可以存取 S-RAM 上的地址变换表,不需要读出闪速存储器的冗余部分。这种情况下,实际的地址变换表 31 以及块状态表 32 例如把 S-RAM 上的特定区域的起始地址为基准点(视为“0000”等),如果从该基准点开始是地址变换表,则可以在成为对应的逻辑块地址值上写入必要的数据(物理块地址),如果是块状态表,则可以在成为对应物理块地址值的位置上写入必要的数据(状态值)。

另外,这时,如果发现逻辑块地址为“FFFF”,即成为消除图形的块则视为队列(Queue)的要素,把该要素登录到队列(Queue)表 50 中。另外,检查该队列的要素内的数据是否全部为“FFFF”,即是否为消除图形。

这种情况下,实际的队列(Queue)表 50,地址变换表 31 以及块状态表 32 例如以 S-RAM 上的特定区域的起始地址为基准点(视为“0000”等),从该基准点开始,在成为分别对应的队列(Queue)数,逻辑块地址值,或者物理块地址值的位置写入必要的数据(分别是物理块地址值,物理块地址值,状态值)。

[读出时的动作]

作为读出时的动作,以图 13 所示的装置中的动作为例进行说明。图 13 是示出闪速存储器系统的一实施例的框图。图示例的闪速存储器系统具有与主计算机 1 连接的 PC 卡总线 2a,与该 PC 卡总线 2a 连接的缓冲器控制单元 35 内的缓冲器 A36 和缓冲器 B37(在该例中作为一体进行表示,然而也可以分别独立地存在)。另外,在上述 PC 卡总线 2a 中,连接着 PCMCIA 块 38 和 ATA 块 39。在上述

缓冲器控制单元 35 内的缓冲器 A36 和缓冲器 B37 中，连接着作为闪速存储器接口 4 的闪速序列发生器 41 和 ECC 控制单元 42（在该例中作为一体进行表示，然而也可以分别独立地存在），在该闪速序列发生器 41 等中连接着闪速存储器 5 的存储器阵列。

另外，控制单元 34 连接着上述缓冲器控制单元 35，PCMCIA 块 38，ATA 块 39，闪速序列发生器 41，把它们综合地进行控制。该控制单元 34 中连接着能够直接存取的 ROM、RAM，使得存储控制算法和运算所必需的数据。另外，地址变换表 31 和块状态表 32 通常形成在上述 RAM 上。

PC 卡总线 2a 是用于连接被称为 PC 卡的卡片形状的扩展系统（例如，SCSI 接口单元，MODEM 单元，存储器卡等）的总线系统，在本例中是通过 PCMCIA 标准的预定卡片槽可以连接 PC 卡的总线。这样的 PC 卡能够特别理想地应用在膝上型计算机等的便携设备和从总线收发信息的系统等中。

缓冲器控制单元 35 以及缓冲器 A36、缓冲器 B37 与上述的数据中继装置 3a 是相同的结构，省略说明。

PCMCIA 块 38 作用为适宜地与上述 PC 卡总线 2a 连接的 PC 卡总线接口。即，PC 卡总线 2a 的控制系统（未图示）存储着用于识别连接了 PC 卡所必需的数据和用于识别所必需的顺序，能够自动地进行必要的操作。ATA 块 39 的工作寄存器具有收容着从主计算机提供的作为同位标磁道组、磁头、扇区的数据的 CHS 地址，或者反之，把这些数据提供给主计算机一侧，用于变换为主计算机一侧的逻辑地址的数据。由此，主计算机 1 能够把闪速存储器系统宛如硬盘一样进行处理。

闪速存储器 5 构成为集合了多个存储器元件的存储器阵列，使得能够确保必要的存储总量。其它的结构与上述的结构例相同，在相同的构成要素上标注相同的符号并且省略说明。

这样，图示例的闪速存储器系统收容在 PC 卡内，使得能够与 SCSI 规格的硬盘相同地进行处理。通过作为 PC 卡能够装卸自由地

与主计算机连接，能够用小型，轻量的卡片自由地移动或者保存比较大容量的数据。而且，由于以往的软件能够与可识别的硬盘相同地进行处理，因此能够灵活地运用已有的软件，处理也很容易。

作为读出时的动作，首先，控制单元 34 通过上述缓冲器 A36(或者 B37) 等接收来自主计算机 1 的地址数据。这时，从主计算机传送来的地址数据如果是 LBA 方式，则通过使用了公式 (I) 的与上述相同的作业，进行向逻辑块地址以及偏移值的变换。

即，如果把从主计算机传送来的逻辑地址记为 LBAs，则逻辑块地址 LBA 用下式求出

$$LBA = LBAs/k$$

k = 最小消除单位内的扇区数 (8 或者 16)

余数 m 成为表示块内的扇区位置的偏移值。

另外，从主计算机传送来的数据是 CHS 方式的情况下，根据以下的计算公式 (II) 变换为 LBA 方式的数据。另外，这样的变换功能也可以像 AT 块等那样，独立设计为具有用于把 CHS 方式的数据变换为 LBA 方式的数据的专用功能的块。

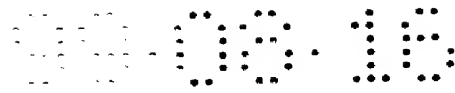
$$LBA = (C \times HpC + H) \times SpH + S - 1 \quad (II)$$

这里，C: 柱面号，H: 磁头号，S: 扇区号，HpC: 磁头/柱面，SpH: 扇区/磁头。

所得到的逻辑块地址通过数据变换表 31 变换为物理块地址，在该地址上加入偏移值成为物理地址数据。

在读出动作中，通常自动地进行闪速存储器 5 的读出和向主计算机 1 的传送。即，如果控制单元 34 设定希望读出的存储器区域的地址，则闪速存储器接口 4 例如闪速序列发生器 41 自动地读出其存储器区域的数据。缓冲器控制单元 35 把传送来的数据与闪速存储器 5 一侧和主计算机 1 一侧的传送速度相吻合顺序地进行发送。

ATA 块 39 如果有来自控制单元 34 的数据传送的指令则解除工作状态，通知主计算机开始传送数据。这样，控制单元 34 在进行了地址变换作业以后，通过进行预定的设定操作，从用于数据传送的



操作中解放出来，能够先行一步进行下一个地址变换作业。另外，由于数据变换作业如上述那样能够高速处理，因此通常在数据传送中结束。由此，格外地提高数据传送速度。

作为写入数据的动作，至地址变换为止的动作与上述读出时的动作相同。接着，进行地址变换，进行向闪速存储器 5 上的预定地址的写入动作。

闪速存储器 5 如上述那样不能够进行改写。从而，在写入动作中伴随着块单位的消除动作。这种情况下，在把预先写入的块内的数据暂时读出了以后也能够消除该数据，加写新的数据。然而，在该方法中在写入之前由于伴随着读出动作，因此存取速度迟缓。

因此，最好灵活地运用包含在上述物理地址区中的逻辑地址区的剩余区域。即，暂时把新的数据写入到剩余区域中任意的块内。而且，把原来的具有写入预定地址的块地址作为消除预定地址。另外，把具有块状态表 32 上的写入预定地址的块地址置换为写入了新的数据的剩余区域中的块的地址。由此，上述剩余区域中的块成为具有写入后的数据的新的块地址。

这种情况下，最好灵活地运用剩余区域中的队列 (Queue)。即，从队列 (Queue) 的起始取出要素，暂时把新的数据写入到该块的预定位置 (用偏移值指定)，作为交换预定块。而且，把原来的作为读入目标的块的逻辑块地址作为消除预定地址。

另外，交换预定块的冗余部分中，写入消除预定块的块地址和状态值。这时，对于地址变换表的逻辑块也可以进行改写。由此，从上述队列 (Queue) 取出的块成为具有写入后的数据的新的逻辑块地址。

而且，最好在全部的或者预定量的数据传送结束的时刻，把上述消除预定块的剩余数据 (没有进行改写的数据) 传送到交换预定块内，把消除预定块进行消除。由此，伴随着写入的读出动作以及消除动作在进行了全部的数据传送后进行，外观上，对于主计算机，通过使读入数据的接收早期完成，提高写入时的动作速度。另外，

说明书附图

图1

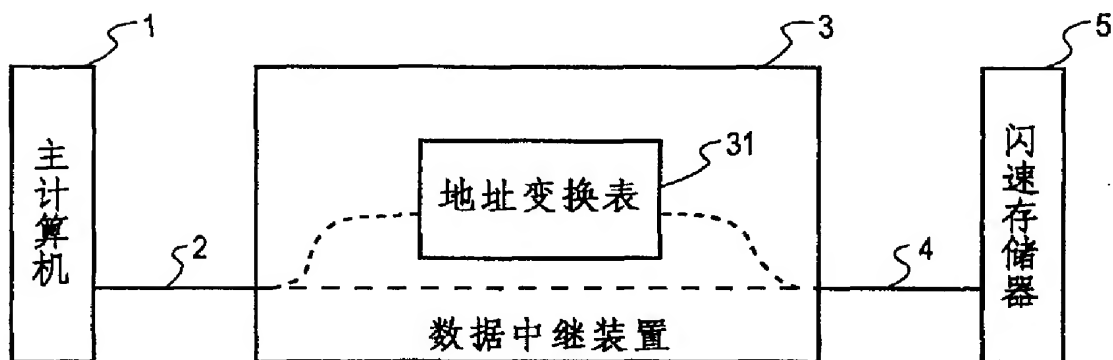


图2 地址变换表

0000	0001	0002	0003	0004	0005	0006	0007
#0000	#0001	#0002	#0004	#0005	#0006	#0007	#0008
0008	0009	0010	0011	0012	0013	0014	0015
#0009	#0010	#0011	#0012	#0013	#0014	#0015	#0016
0016	0017	0018	0019	0020	0021	0022	0023
#0017	#0018	#0019	#0021	#0022	#0023	#0024	#0025
0024	0025	0026	0027	0028	0029	0030	0031
#0026	#0027	#0028	#0029	#0030	#0031	#0032	#0033
0032	0033	0034	0035	0036	0037	0038	0039
#0034	#0035	#0036	#0037	#0038	#0039	#0040	#0041

Nh-7	Nh-6	Nh-5	Nh-4	Nh-3	Nh-2	Nh-1	Nh
#Nh-5	#Nh-4	#Nh-3	#Nh-2	#Nh-1	#Nh	#Nh+1	#Nh+2

图 3

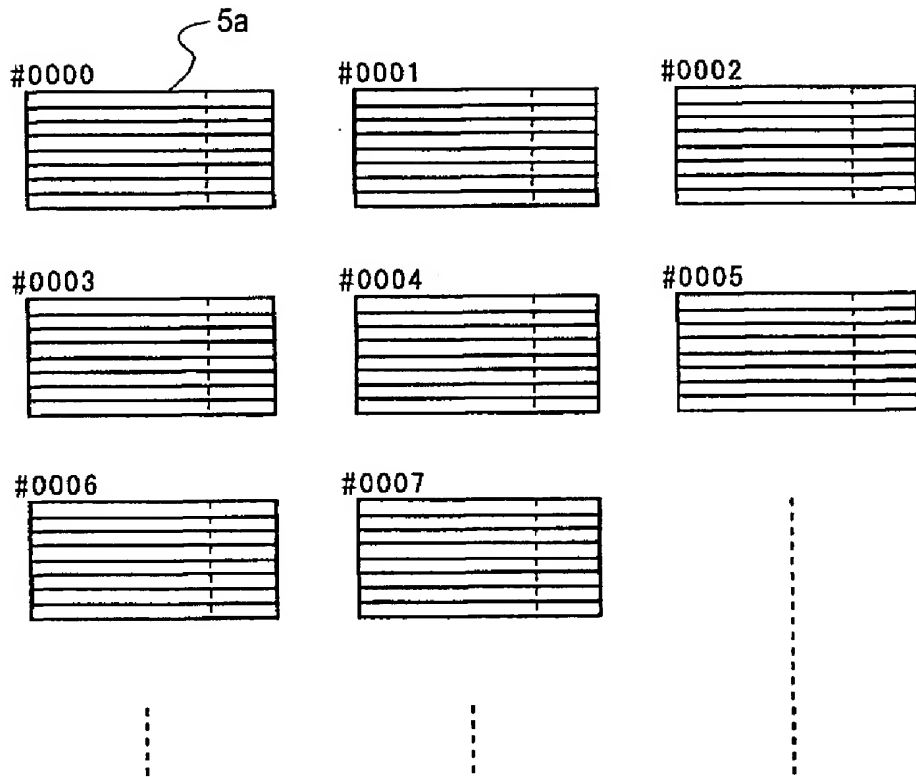


图 4

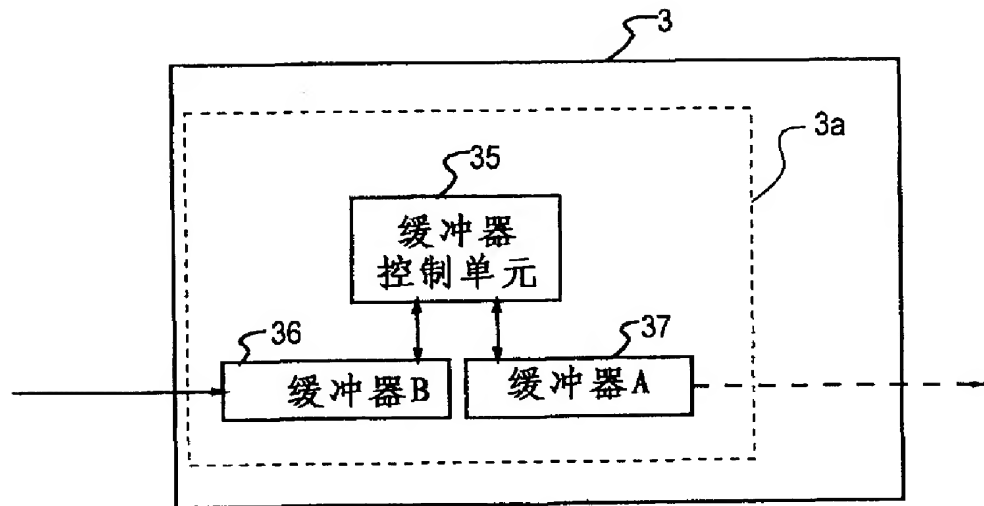


图 5 块状态表

32

#0000 FF	#0001 FF	#0002 FF	#0003 FF	#0004 FF	#0005 FF	#0006 FF	#0007 FF
#0008 FF	#0009 FF	#0010 FF	#0011 FF	#0012 FF	#0013 FF	#0014 FF	#0015 FF
#0016 FF	#0017 FF	#0018 FF	#0019 FF	#0020 FF	#0021 FF	#0022 FF	#0023 FF
#0024 FF	#0025 FF	#0026 FF	#0027 FF	#0028 FF	#0029 FF	#0030 FF	#0031 FF
#0032 FF	#0033 FF	#0034 FF	#0035 FF	#0036 FF	#0037 FF	#0038 FF	#0039 FF

#Nmax-7 FF	#Nmax-6 FF	#Nmax-5 FF	#Nmax-4 FF	#Nmax-3 FF	#Nmax-2 FF	#Nmax-1 FF	#Nmax FF
---------------	---------------	---------------	---------------	---------------	---------------	---------------	-------------

图6

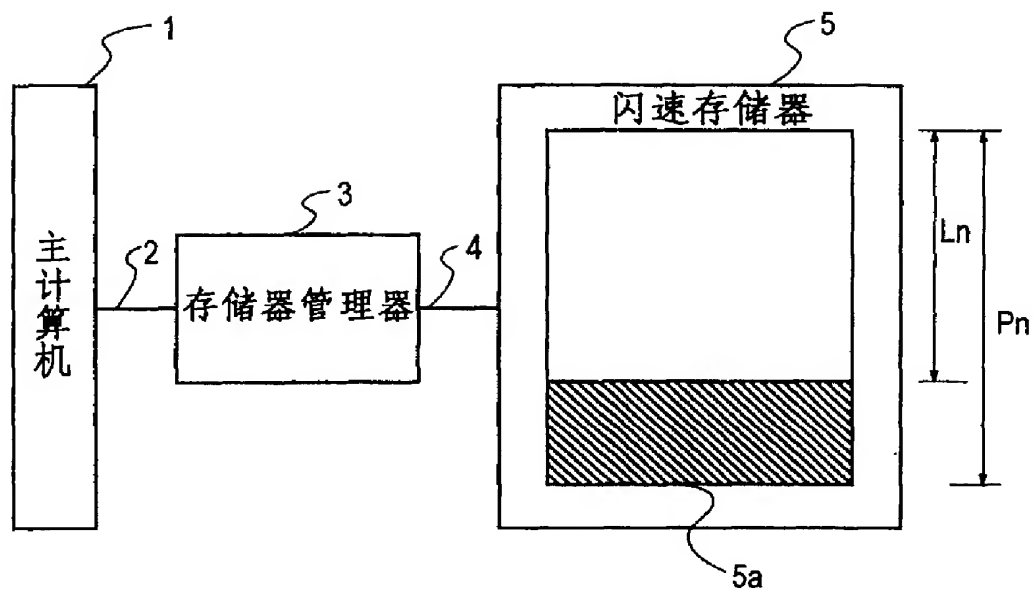


图7

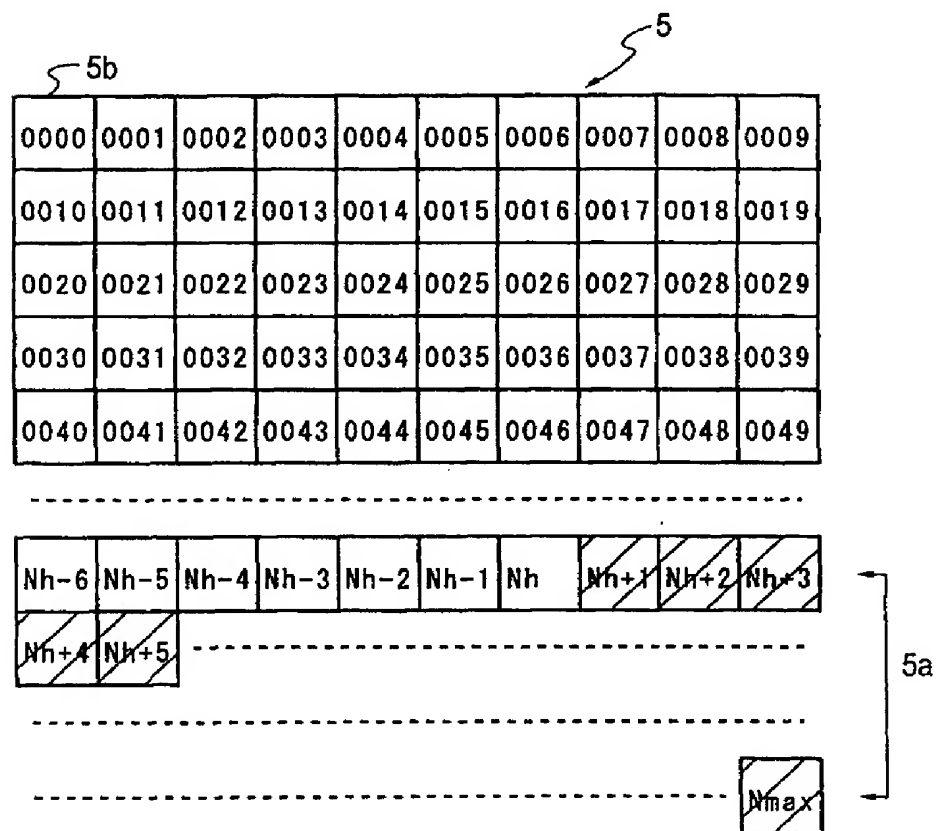


图 9

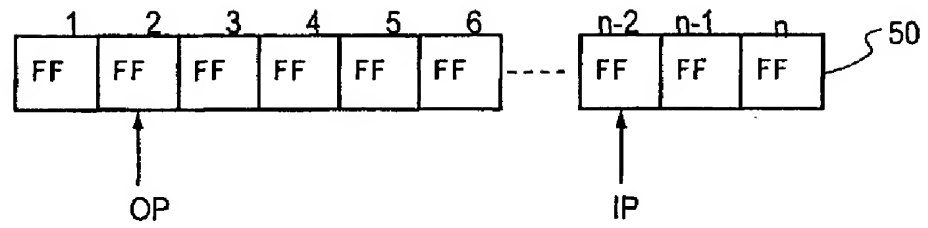


图 10

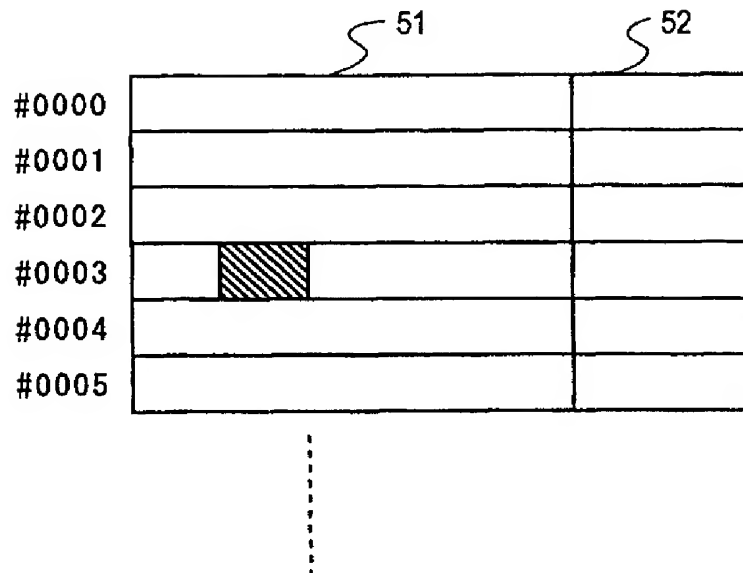


图 11


32

#0000 FF	#0001 FF	#0002 FF	#0003 OF	#0004 FF	#0005 FF
#0006 FF	#0007 FF	#0008 FF	#0009 FF	#0010 FF	#0012 FF
#0013 FF	#0014 FF	#0015 FF	#0016 FF	#0017 FF	#0018 FF
#0019 FF	-----				

⋮

图 12

51 52

#0000		FF 00
#0001		FF 01
#0002		FF 02
#0003		OF --
#0004		FF 03
#0005		FF 04

⋮

#0500		FF Nh-1
#0501		FF Nh-2
#0502		
#0503		

图13

